

## Practical Task 1: Upload and Retrieve Files with Azure Blob Storage

Use Azure Blob Storage for storing and retrieving files.

### Requirements:

1. Create a new storage account in Azure.
2. Set up a Blob container named "my-container" with public access.
3. Upload a sample text file to the Blob container using the Azure portal.
4. Download the uploaded file to verify successful retrieval.
5. Use Azure Storage Explorer to manage and view blobs in your container.

### Actions Taken:

#### 1. Created a New Storage Account:

The screenshot shows the Microsoft Azure Resource groups interface. A new resource group named 'LessonStorage1' is being created. In the 'Resources' section, a storage account named 'mystorageaccount1232' is listed under the 'Type equals all' filter. The storage account has a 'Name' of 'mystorageaccount1232', a 'Type' of 'Storage account', a 'Location' of 'West Europe', and a 'Status' of 'Succeeded'.

#### 2. Set Up a Blob Container and uploaded a file forContainer.txt:

The screenshot shows the Microsoft Azure Storage Explorer interface. A blob container named 'my-container' is selected. A file named 'forContainer.txt' is listed in the blob list. The file details are: Name: 'forContainer.txt', Modified: '1/17/2025, 7:39:32 PM', Access tier: 'Hot (inferred)', Archive status: 'Not yet archived', Blob type: 'Block blob', Size: '37.8', Lease state: 'Available'.

### 3.Verified File Retrieval:

The screenshot shows the Microsoft Azure Storage Explorer interface. On the left, there's a sidebar for the 'my-container' blob storage, which includes a search bar, an 'Upload' button, and a 'Change access level' option. Below these are sections for 'Diagnose and solve problems', 'Access Control (IAM)', and 'Settings'. A 'Search blobs by prefix (case-sensitive)' input field and a 'Show deleted blobs' toggle switch are also present. On the right, the main pane displays a blob named 'forContainer.txt'. The blob details show it was created on '11/10/2023' at '11:45 AM' with a size of '1.00 KB'. The blob content is a single line of text: 'Hello Azure and Vivi to Blob Storage'. Below the blob details, there are tabs for 'Overview', 'Versions', 'Snapshots', 'Edit', and 'Generate SAS'.

### 4.Managed Blobs with Azure Storage Explorer:

This screenshot shows the Microsoft Azure Storage Explorer interface integrated into a web browser window. The left side is the same Azure storage container overview as the previous screenshot. The right side is the Storage Explorer interface, which has a toolbar with various file operations like Upload, Download, Open, Preview, New Folder, Select All, Copy, Paste, Clone, and More. The 'EXPLORER' panel on the left shows a tree view of storage accounts, containers, and blobs. The 'my-container' blob is selected. The main pane shows a table of blobs with columns for Name, Access Tier, and Last Mod. There is one entry: 'forContainer.txt' with an 'Access Tier' of 'Hot (inferred)' and a 'Last Mod' date of '11/10/2023'.

#### Practical Task 2: Lifecycle Management for Blob Storage

Implement lifecycle management policies to optimize storage costs.

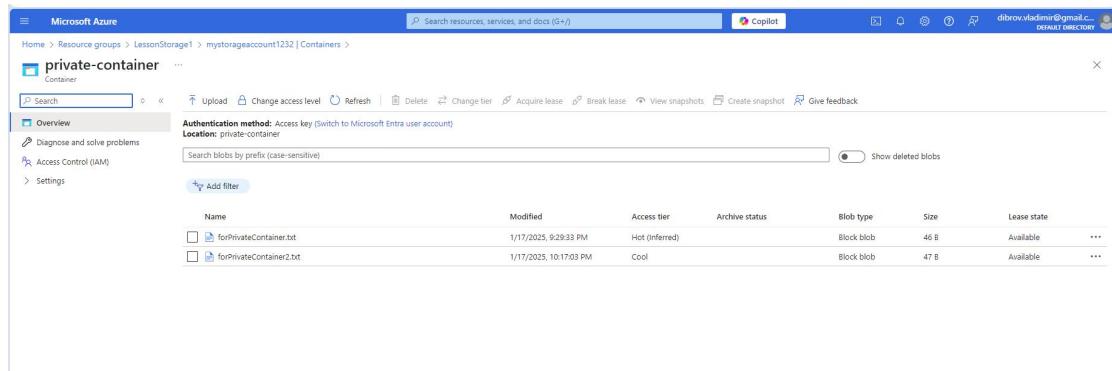
##### Requirements:

1. Create a storage account and a Blob container named "lifecycle-container."
2. Upload multiple files of varying sizes to the container.
3. Create a lifecycle management policy to move blobs to the Cool tier after 30 days and delete blobs older than 90 days.
4. Simulate policy execution by manually testing with different file creation timestamps.
5. Verify that blobs are moved or deleted according to the policy

## Actions Taken:

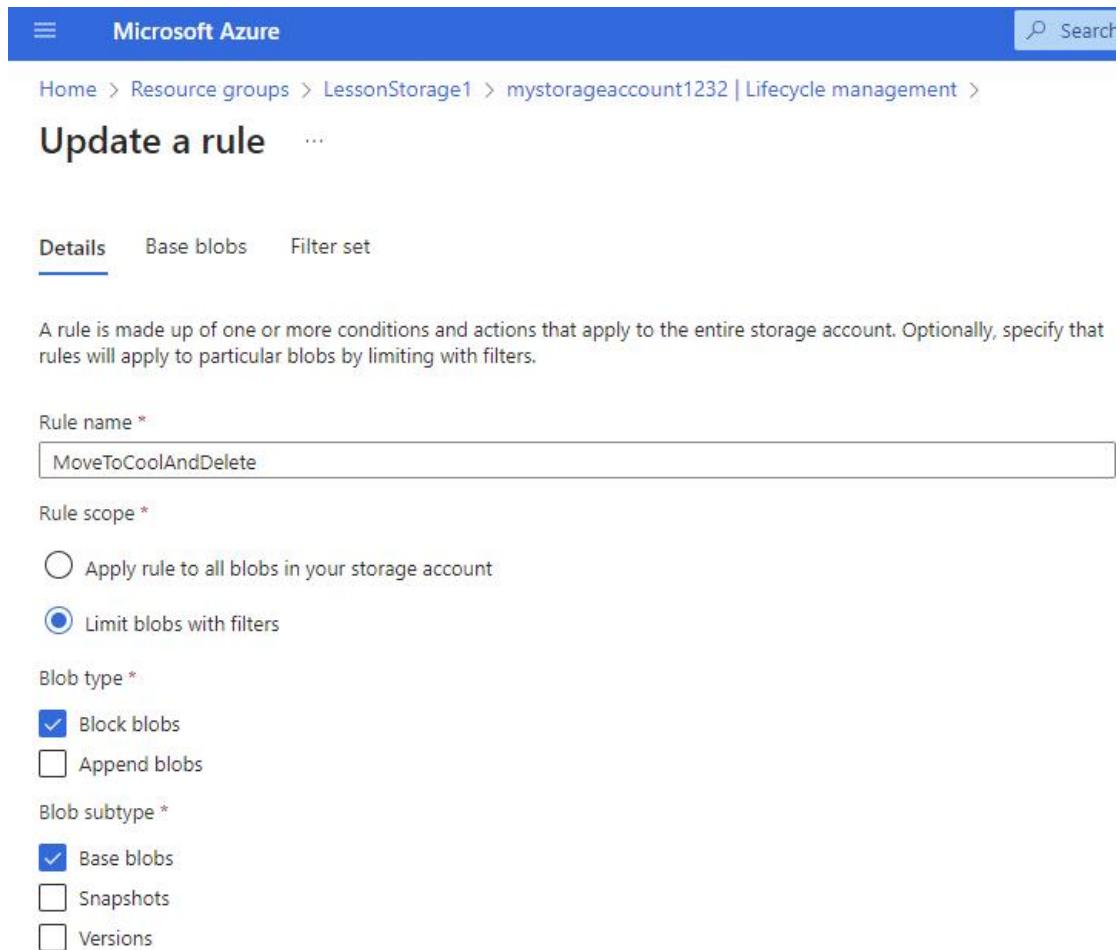
### Created a Storage Account and Container:

#### 1. Uploaded Data:



The screenshot shows the Microsoft Azure Storage Container Overview page for a container named "private-container". The container has two blobs: "forPrivateContainer.txt" (Modified: 1/17/2025, 9:29:33 PM, Access tier: Hot (inferred), Blob type: Block blob, Size: 46 B) and "forPrivateContainer2.txt" (Modified: 1/17/2025, 10:17:03 PM, Access tier: Cool, Blob type: Block blob, Size: 47 B). The interface includes tabs for Overview, Diagnose and solve problems, Access Control (IAM), and Settings. There are also buttons for Upload, Delete, Change tier, Acquire lease, Break lease, View snapshots, Create snapshot, and Give feedback.

#### 2. Defined Lifecycle Management Policy:



The screenshot shows the Microsoft Azure Lifecycle management rule configuration page. A rule named "MoveToCoolAndDelete" is being updated. The rule applies to all blobs in the storage account. It moves blobs to the Cool access tier and then deletes them. The rule is set to run daily at 00:00 UTC. The configuration includes sections for Details, Base blobs, and Filter set. Under Details, it specifies the rule name "MoveToCoolAndDelete" and the rule scope "Apply rule to all blobs in your storage account". Under Base blobs, it selects "Block blobs". Under Filter set, it selects "Base blobs".

Details    Base blobs    Filter set

A rule is made up of one or more conditions and actions that apply to the entire storage account. Optionally, specify that rules will apply to particular blobs by limiting with filters.

Rule name \*

MoveToCoolAndDelete

Rule scope \*

Apply rule to all blobs in your storage account

Limit blobs with filters

Blob type \*

Block blobs

Append blobs

Blob subtype \*

Base blobs

Snapshots

Versions

Microsoft Azure

Search resources, services

Home > Resource groups > LessonStorage1 > mystorageaccount1232 | Lifecycle management >

## Update a rule

Details    Base blobs    Filter set

Lifecycle management uses your rules to automatically move blobs to cooler tiers or to delete them. If you create multiple rules, the associated actions must be implemented in tier order (from hot to cool storage, then archive, then deletion).

```
graph TD; A[If Base blobs haven't been modified in 30 days] --> B[Then Move to cool storage]; C[If Base blobs haven't been modified in 90 days] --> D[Then Delete the blob]; E[Add conditions]
```

If Base blobs haven't been modified in 30 days

Then Move to cool storage

If Base blobs haven't been modified in 90 days

Then Delete the blob

+ Add conditions

Simulation “moving blobs to the Cool”

Microsoft Azure

Search resources, services, and docs

Home > Resource groups > LessonStorage1 > mystorageaccount1232 | Lifecycle management >

## Update a rule

Details    Base blobs    Filter set

Blob prefix

Filter blobs by name or first letters. To find items in a specific container, enter the name of the container followed by a forward slash, then the blob name or first letters. For example, to show all blobs starting with "a", type: "mycontainer/a".

Enter a prefix or file path such as "mycontainer/prefix"

Blob index match

If you have indexed items in containers with keys and values, you can filter for them.

Key	Value
customDate	* == 2024-11-15T10:00:00Z
Enter an index key	== Enter a value

The screenshot shows the Microsoft Azure Storage Explorer interface. On the left, the Explorer sidebar lists storage accounts, blob containers, file shares, queues, and tables. The 'private-container' blob container is selected. In the main pane, a table lists blobs with columns for Name, Access Tier, and Last Modified. Two blobs are present: 'forPrivateContainer.txt' (Hot inferred) and 'forPrivateContainer2.txt' (Cool). A 'Properties' dialog is open for 'forPrivateContainer2.txt', displaying its name, URL, Blob Type, Container, Last Modified, and Created On. The URL is https://mystorageaccount123.blob.core.windows.net/private-container/forPrivateContainer2.txt. The blob type is BlockBlob, and it was created on 2025-01-17T20:14:53.000Z.

### 3. Simulation for deleting :

```
PS C:\Users\User>
PS C:\Users\User> az storage blob metadata update ` 
>>   --container-name private-container ` 
>>   --name forPrivateContainer2.txt ` 
>>   --account-name mystorageaccount123 ` 
>>   --metadata customDate="2023-10-01T00:00:00Z"

There are no credentials provided in your command and environment, we will query for account key for your storage account.
It is recommended to provide --connection-string, --account-key or --sas-token in your command as credentials.

You also can add '--auth-mode login' in your command to use Azure Active Directory (Azure AD) for authorization if your login account is assigned required RBAC roles.
For more information about RBAC roles in storage, visit https://docs.microsoft.com/azure/storage/common/storage-auth-aad-rbac-cli.

In addition, setting the corresponding environment variables can avoid inputting credentials in your command. Please use --help to get more information about environment variable usage.
```

Home >

## Update a rule

...

Details    Base blobs    Filter set

### Blob prefix

Filter blobs by name or first letters. To find items in a specific container, enter the name of the container followed by a forward slash, then the blob name or first letters. For example, to show all blobs starting with "a", type: "mycontainer/a".

### Blob prefix

Enter a prefix or file path such as "mycontainer/prefix"

### Blob index match

If you have indexed items in containers with keys and values, you can filter for them.

#### Key

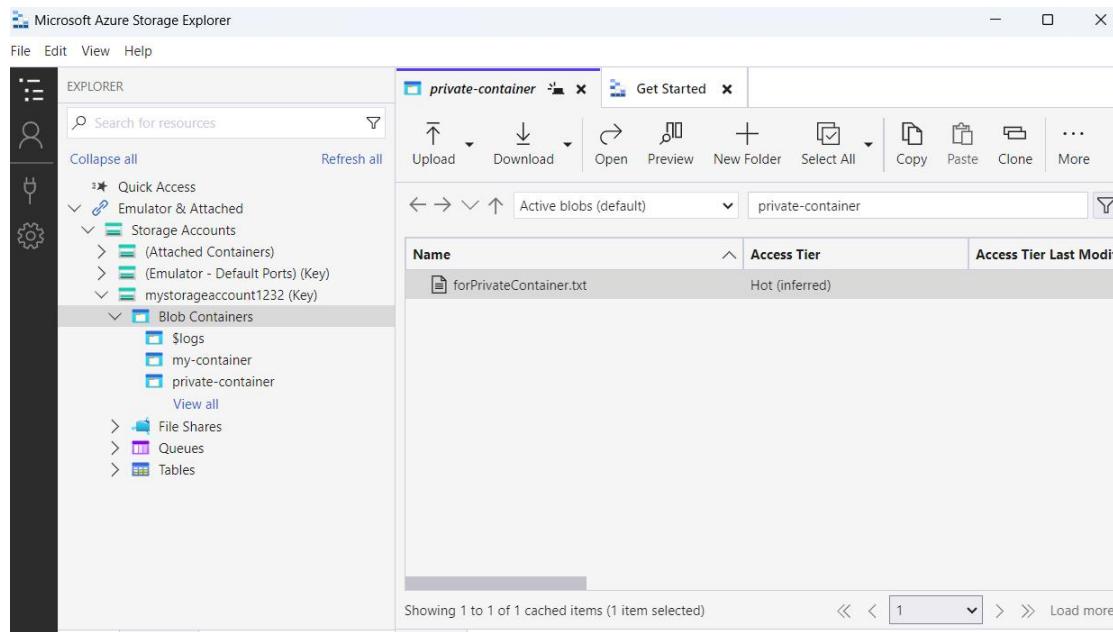
#### Value

customDate \* == 2023-10-01T00:00:00Z

Enter an index key == Enter a value

## Monitored Execution:

1. Enabled monitoring via the **Activity Log** in Azure Portal to track lifecycle policy application.
2. Verified that blobs transitioned to the Cool tier and older blobs were deleted as per the policy.



## Practical Task 3: Implementing an Azure Queue for Message Storage

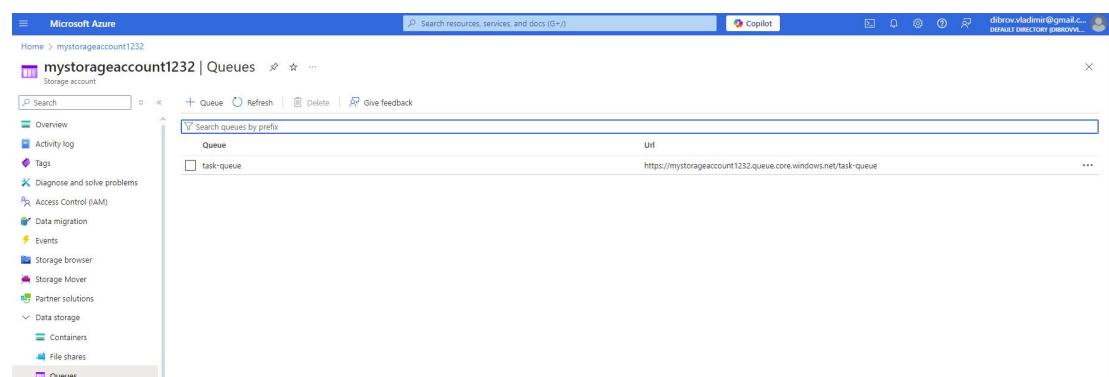
Create and manage an Azure Queue to store and process messages.

**Requirements:**

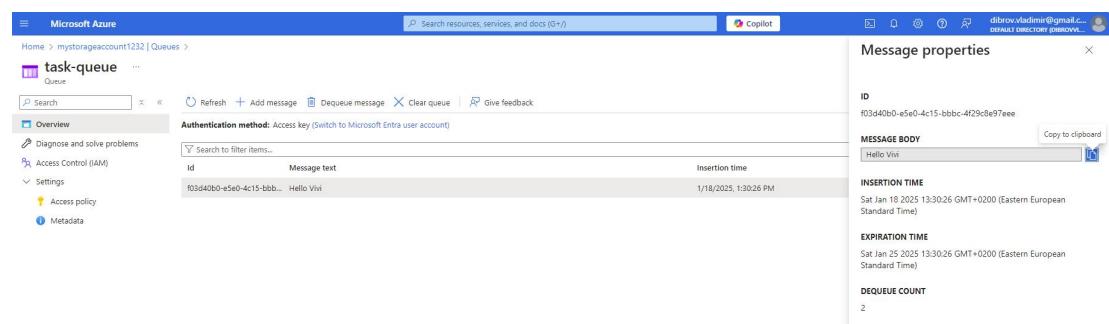
1. Create a storage account and enable the Queue service.
2. Create a queue named "task-queue."
3. Use Azure Storage Explorer or the Azure portal to add messages to the queue.
4. Retrieve and process messages directly using Azure Storage Explorer or the Azure portal interface.
5. Verify that processed messages are removed from the queue manually.

**Actions Taken:**

1. Created Queue in mystorageaccount1232 :



2. Added message:



3. Received message and decoding content :

```
PS C:\Users\User> az storage message get `>> --queue-name task-queue`>> --account-name mystorageaccount1232Command group 'storage message' is in preview and under development. Reference and support levels: https://aka.ms/CLI_refstatusThere are no credentials provided in your command and environment, we will query for account key for your storage account. It is recommended to provide --connection-string, --account-key or --sas-token in your command as credentials. You also can add '--auth-mode login' in your command to use Azure Active Directory (Azure AD) for authorization if your login account is assigned required RBAC roles. For more information about RBAC roles in storage, visit https://docs.microsoft.com/azure/storage/common/storage-auth-aad-rbac-cli.In addition, setting the corresponding environment variables can avoid inputting credentials in your command. Please use --help to get more information about environment variable usage.[{  "content": "SGVsbG9vVmL2aQ==",  "dequeueCount": 2,  "expirationTime": "2025-01-25T11:30:26+00:00",  "id": "f03d40b0-e5e0-4c15-bbbc-4f29c9e97eee",  "insertionTime": "2025-01-18T11:30:26+00:00",  "popReceipt": "AgAAAAAAA=AAA+ueP1p2wE=",  "timeNextVisible": "2025-01-18T11:36:38+00:00"}]PS C:\Users\User>
```

Do you have to deal with **Base64** format? Then this site is perfect for you! Use our

### Decode from Base64 format

Simply enter your data then push the decode button.

**SGVsbG8gVmI2aQ==**

**For encoded binaries (like images, documents, etc.) use the file upload form a little further down.**

Source character set: **UTF-8**

Decode each line separately (useful for when you have multiple entries).

Live mode OFF Decodes in real-time as you type or paste (supports only the UTF-8 character set).

**DECODE** Decodes your data into the area below.

**Hello Vivi**

Microsoft Azure Storage Explorer

File Edit View Help

EXPLORER

Search for resources

Collapse all Refresh all

- Quick Access
- Emulator & Attached
- Storage Accounts
  - (Attached Containers)
  - (Emulator - Default Ports) (Key)
  - mystorageaccount1232 (Key)**
    - Blob Containers
      - \$logs
      - my-container
      - private-container
    - File Shares
    - Queues
      - task-queue**
  - Tables

task-queue Get Started mystorageaccount1232 \$logs

Add Import Export Dequeue Delete Clear Queue Requeue Move More

ID	Message Text	Insertion Time
f03d40b0-e5e0-4c15-bbbc-4f29c8e97eee	Hello Vivi	1/18/2025 1:30:45 PM

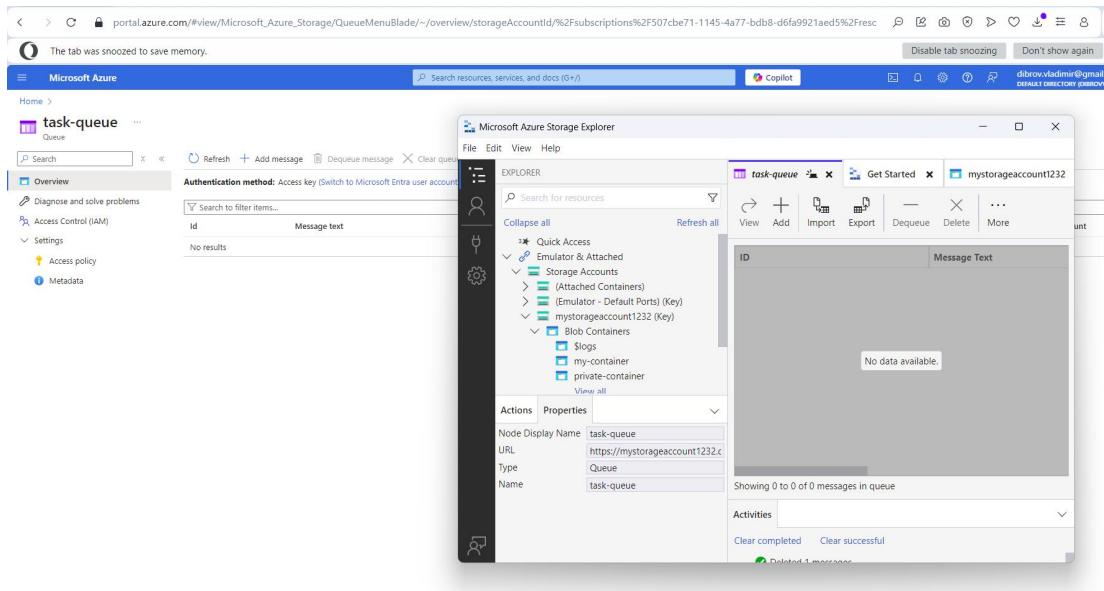
Showing 1 to 1 of 1 messages in queue

Activities

Clear completed Clear successful

Added message successfully  
Finished dequeuing message  
Deletion of 'kvibcrap.txt' from 'private-container/' completed: 1

## Removed from the queue manually



## Practical Task 4: Configuring Azure File Share and Mounting on a Local Machine

Set up and access Azure File Storage for shared file access.

### Requirements:

1. Create a storage account and set up an Azure File Share.
2. Upload a file to the file share using the Azure portal.
3. Generate a connection script for Windows or Linux and use it to mount the file share on your local machine.
4. Verify the mounted file share and ensure it displays the uploaded file.
5. Add another file to the file share from the local machine and confirm it reflects in Azure.

## Actions Taken

## 1. Set up an Azure File Share in mystorageaccount1232 :

The screenshot shows the Azure portal interface for creating a new file share. The top navigation bar includes 'Microsoft Azure', a search bar, and a 'Copilot' button. The main title is 'myfileshare | SMB File share'. The left sidebar has sections like 'Overview', 'Diagnose and solve problems', 'Access Control (IAM)', 'Browse', 'Operations', and 'Solutions'. The 'Overview' tab is selected. Key details shown include:

Setting	Value	Notes
Storage account	mystorageaccount1232	Share URL
Resource group (move)	LessonStorage1	Redundancy
Location	West Europe	Configuration modified
Primary/Secondary location	Primary: West Europe, Secondary: North Europe	
Subscription (move)	Azure subscription 1	
Subscription ID	507cbe71-1145-4a77-bdb8-d6fa9921aed5	

Below the properties, there are tabs for 'Properties', 'Capabilities (2)', and 'Tutorials'. On the right, there are sections for 'Feature status' (Soft delete, Large file shares), 'Identity-based access' (Directory service, Domain), and 'SMB protocol settings'.

## 2. Uploaded file :

The screenshot shows the 'myfileshare | Browse' page. The left sidebar includes 'Overview', 'Diagnose and solve problems', 'Access Control (IAM)', 'Browse' (selected), 'Operations', 'Solutions', and 'Backup'. The main area displays a table of files:

Name	Type	Size	Actions
forFileShare.txt	File	34 B	...

## 3. Generated a connection script for Windows and used it to mount the file share

The screenshot shows the Microsoft Azure portal interface. On the left, under 'Storage file shares', there is a 'myfileshare' entry. On the right, a 'Connect' dialog box is open for this share. It contains a PowerShell window with the following script:

```

PS C:\Users\user> $connectTestResult = Test-NetConnection -ComputerName mystora
geaccount123.file.core.windows.net -Port 445
WARNING: TCP connect to 20.209.73.16 : 445) failed
WARNING: Ping to 20.209.73.16 failed with status: DestinationHostUnreachable
PS C:\Users\user> if ($connectTestResult.TcpTestSucceeded) {
    >>     # Save the password so the drive will persist on reboot
    >>     cmdkey /add:"mystorageaccount123.file.core.windows.net" /u:"$env:username" /p:"$env:password"
    >>     New-PSDrive -Name Z -PSProvider FileSystem -Root "\\mystorageaccount123
    .file.core.windows.net\myfileshare" -Persist
    >> } else {
    >>     Write-Error "Unable to reach the Azure storage account via port
    445. Check to make sure your organization or ISP is not blocking port 445, or
    use Azure P2S VPN, Azure S2S VPN, or Express Route to tunnel SMB traffic over a
    different port."
    >> }

```

The 'Connect' dialog also displays configuration options for connecting from Windows, Linux, or macOS, including authentication methods like Active Directory or Storage account key.

The screenshot shows the Microsoft Azure portal interface under 'Virtual network gateways'. A 'VnToFile' gateway is selected. In the 'Point-to-site configuration' section, the following settings are visible:

- Address pool:** 172.16.0.0/24
- Tunnel type:** OpenVPN (SSL)
- Authentication type:** Azure certificate
- Root certificates:** A certificate named 'root' is listed with its public certificate data shown.
- Allocated IP addresses:** A dropdown menu for selecting an IP address.

I tried to connect to Azure File Share via SMB protocol using VPN.

### Connection Testing:

Used Test-NetConnection, but port 445 was blocked.

### Attempted via Azure VPN:

Configured virtual network gateway and point-to-site VPN.

Error while connecting: Server not responding.

### Firewall Configuration:

Added allow rule for port 445, but connection still fails.

The primary issue appears to be port 445 being blocked at the ISP or network level.

## Practical Task 5: Storing and Querying Data with Azure Table Storage

Set up and use Azure Table Storage for structured data.

### Requirements:

1. Create a storage account and enable the Table service.
2. Create a table named "employee-data".
3. Add sample data (e.g., employee IDs, names, and roles) to the table using Azure Storage Explorer or Azure CLI.
4. Query the table for specific data using filters (e.g., "Role = Developer").
5. Delete specific entries from the table and verify the changes.

### Actions Taken:

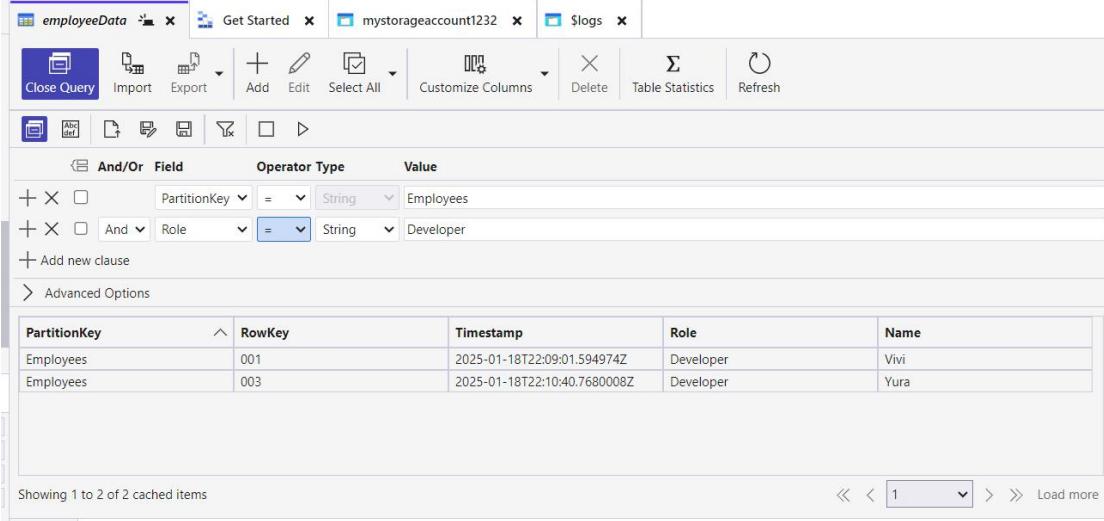
1:Created table "employeeData" in storage "mystorageaccount1232"

PartitionKey	RowKey	Timestamp	Role	Name
Employees	001	2025-01-18T22:09:01.594974Z	Developer	Vivi
Employees	002	2025-01-18T22:10:00.838907Z	Front end	Ira
Employees	003	2025-01-18T22:10:40.7680008Z	Developer	Yura

2.Added data to the table

PartitionKey	RowKey	Timestamp	Role	Name
Employees	001	2025-01-18T22:09:01.594974Z	Developer	Vivi
Employees	002	2025-01-18T22:10:00.838907Z	Front end	Ira
Employees	003	2025-01-18T22:10:40.7680008Z	Developer	Yura

### 3.Querying a table for specific data using filters and PowerShell



The screenshot shows the Azure Storage Explorer interface with the 'employeeData' table selected. The query builder at the top has two clauses: 'PartitionKey = String Employees' and 'Role = String Developer'. The results table below shows two items: one for 'Employees' with RowKey '001' and another for 'Employees' with RowKey '003'. Both entries have the 'Role' field set to 'Developer' and the 'Name' field set to either 'Vivi' or 'Yura'.

PartitionKey	RowKey	Timestamp	Role	Name
Employees	001	2025-01-18T22:09:01.594974Z	Developer	Vivi
Employees	003	2025-01-18T22:10:40.7680008Z	Developer	Yura

Showing 1 to 2 of 2 cached items

```
PS C:\Users\user> az storage entity query --connection-string "DefaultEndpointsProtocol=https;AccountName=mystorageaccount1232;AccountKey=4Ai4yLSW/BK9qm#8iTTF7872+XnB8oz8XkX060k42k/uF704LEmBZQEm3WSQCvshvUtzFPn4EC7+ASoGzZw==;EndpointSuffix=core.windows.net" --table-name employeeData --filter "Role eq 'Developer'"  
{  
    "items": [  
        {  
            "Name": "Vivi",  
            "PartitionKey": "Employees",  
            "Role": "Developer",  
            "RowKey": "001",  
            "Timestamp": "2025-01-18T22:09:01.594974+00:00",  
            "etag": "W/\\"datetime'2025-01-18T22:09:01.594974Z'\\\""  
        },  
        {  
            "Name": "Yura",  
            "PartitionKey": "Employees",  
            "Role": "Developer",  
            "RowKey": "003",  
            "Timestamp": "2025-01-18T22:10:40.768000+00:00",  
            "etag": "W/\\"datetime'2025-01-18T22:10:40.7680008Z'\\\""  
        }  
}
```

### 4.Deleted certain records from a table using PowerShell

```

PS C:\Users\User> az storage entity delete --connection-string "DefaultEndpointsProtocol=https;AccountName=mystorageaccount1232;AccountKey=4Ai4yLSw/BK9qm48i
Tf7f87Zz+XnB8oz8XkX060k42k/uF704LEmBZQEm3W5QCvshvUttZfPh4EC7+AS0glzz==;EndpointSuffix=core.windows.net" --table-name employeedata --partition-key Employees
--row-key 001
{
  "deleted": null
}
PS C:\Users\User> | 

PS C:\Users\User> az storage entity delete --connection-string "DefaultEndpointsProtocol=https;AccountName=mystorageaccount1232;AccountKey=4Ai4yLSw/BK9qm48i
Tf7f87Zz+XnB8oz8XkX060k42k/uF704LEmBZQEm3W5QCvshvUttZfPh4EC7+AS0glzz==;EndpointSuffix=core.windows.net" --table-name employeedata --partition-key Employees
--row-key 001
{
  "deleted": null
}
PS C:\Users\User> az storage entity query --connection-string "DefaultEndpointsProtocol=https;AccountName=mystorageaccount1232;AccountKey=4Ai4yLSw/BK9qm48i
Tf7f87Zz+XnB8oz8XkX060k42k/uF704LEmBZQEm3W5QCvshvUttZfPh4EC7+AS0glzz==;EndpointSuffix=core.windows.net" --table-name employeedata
{
  "items": [
    {
      "Name": "Ira",
      "PartitionKey": "Employees",
      "Role": "Front end",
      "RowKey": "002",
      "Timestamp": "2025-01-18T22:10:00.838907+00:00",
      "etag": "W/\\"datetime'2025-01-18T22%3A10%3A00.838907Z\\\""
    },
    {
      "Name": "Yura",
      "PartitionKey": "Employees",
      "Role": "Developer",
      "RowKey": "003",
      "Timestamp": "2025-01-18T22:10:40.768000+00:00",
      "etag": "W/\\"datetime'2025-01-18T22%3A10%3A40.768000Z\\\""
    }
  ],
  "nextMarker": {}
}
PS C:\Users\User> |

```

## Practical Task 6: Configuring Shared Access Signatures (SAS) for Secure Access

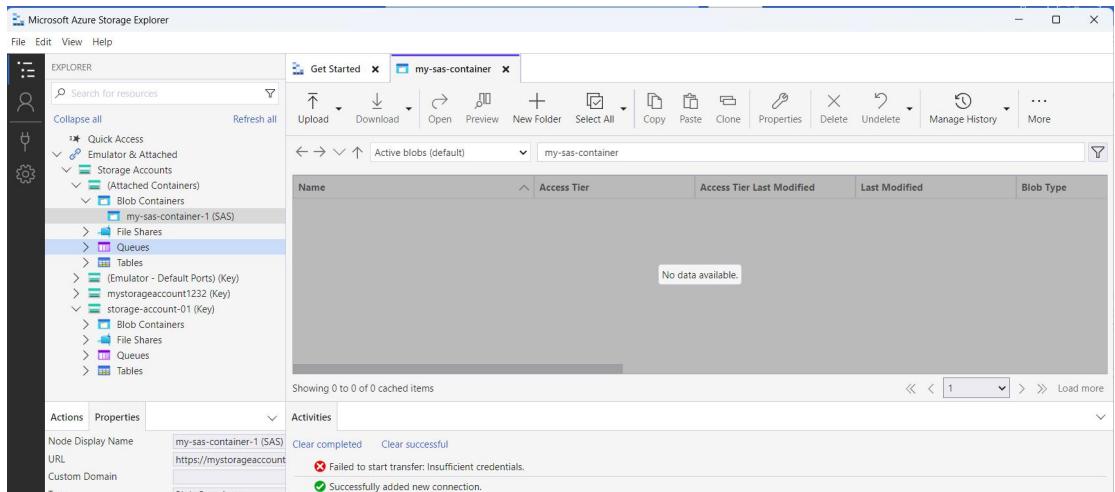
Secure Azure Storage services using SAS tokens.

### Requirements:

1. Create a storage account with Blob, File, Queue, and Table services enabled.
2. Generate a Shared Access Signature (SAS) token for Blob storage with limited permissions (e.g., read-only access).
3. Share the SAS token URL and verify access to the Blob container with the token.
4. Repeat the process for File, Queue, and Table services with different permissions.
5. Analyze the security implications of SAS tokens and expiry times.

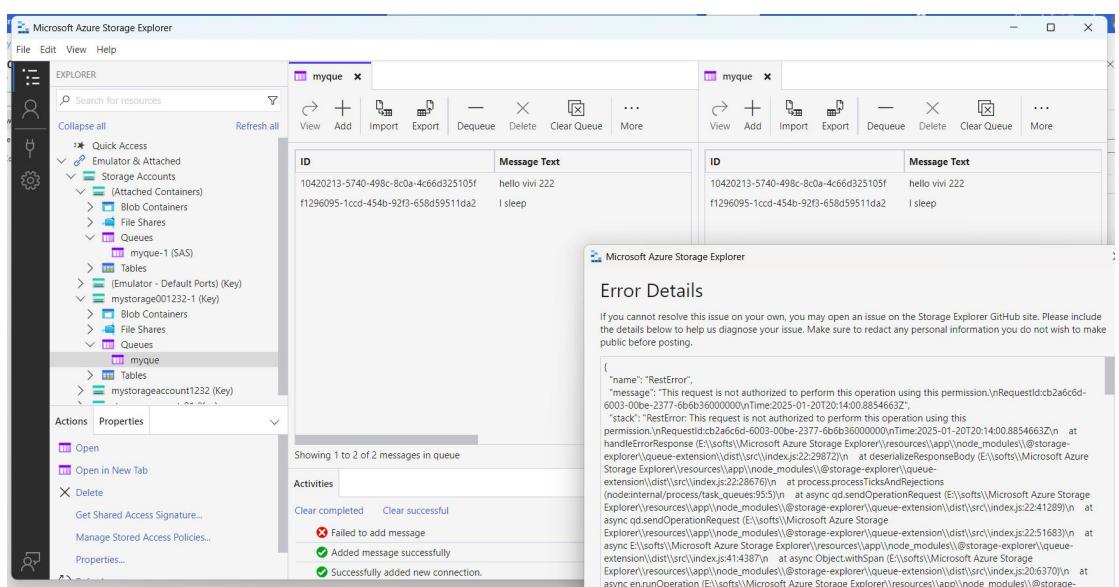
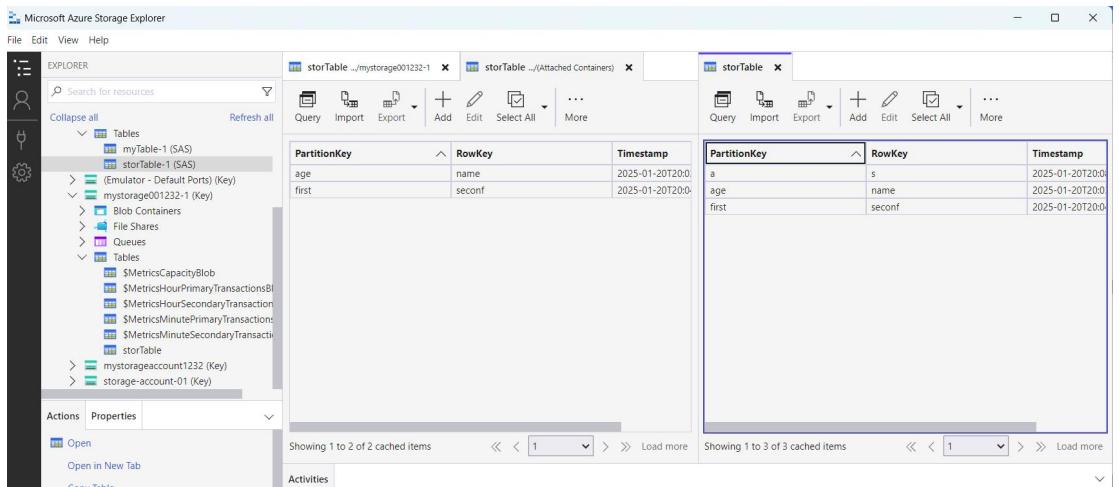
### Actions Taken:

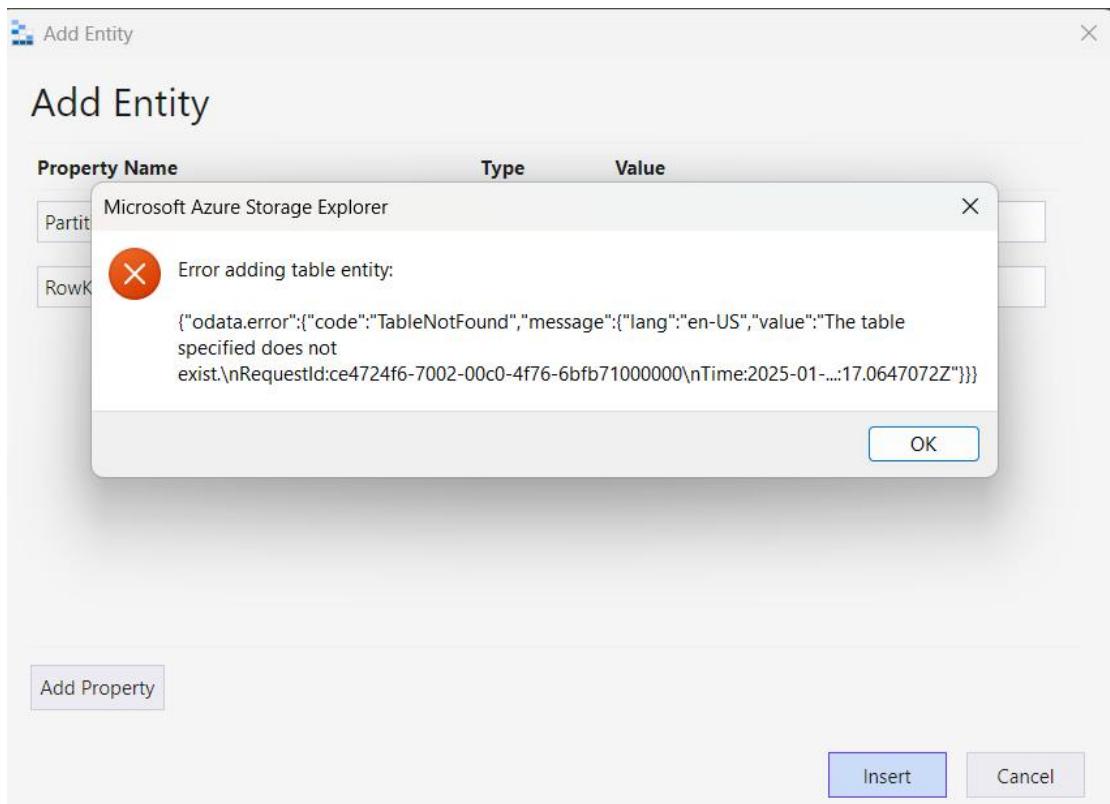
1. Created a storage account with Blob, File, Queue, and Table services enabled.



The screenshot shows the Azure portal's 'Containers' blade for the storage account 'mystorageaccount012342'. It lists two containers: '\$logs' and 'my-sas-container'. The 'my-sas-container' has a last modified time of 1/20/2025, 7:55:47 PM and is set to Private. On the right side, there is a 'Generate SAS' panel where a SAS token is being created for the 'my-sas-container'.

The screenshot shows the Microsoft Azure Storage Explorer interface. The left sidebar shows a tree view of storage accounts and containers. In the center, a modal dialog titled 'Add Entity' is open, showing an error message: 'Error adding table entity: {"odata.error":{"code":"AuthorizationResourceTypeMismatch","message":{"lang":"en-US","This request is not authorized to perform this operation using this resource type.\nRequestId:58455aaa-7002-003f-496f-6b51a6000000\nTime:2025-01-28T02:58:39Z"}}}'. The main pane shows a list of blobs in the 'my-sas-container'.





## 5. My analyzing for the security implications of SAS tokens and expiry times.

- 1.always limit the actions of the period.
- 2.Restrict IP addresses for added security.

### My recommendation:

1. Generate short-term tokens.
2. Use the principle of least privilege (do not grant unnecessary privileges).
3. Store SAS keys in a secure storage (e.g. Azure Key Vault).

## Practical Task 7: Implementing Security Best Practices with Azure RBAC and Managed Identities

Securely manage access to Azure resources and integrate services using Managed Identities.  
**Requirements:**

1. **Configure Azure RBAC for a Storage Account:**
  - o Create a storage account named "secure-storage".
  - o Add a user or service principal with **Storage Blob Data Contributor** role.
  - o Verify that the user or service principal can upload and download blobs to the account.
  - o Attempt access with an unauthorized user and verify access is denied.
2. **Set Up a Managed Identity for an Azure Virtual Machine:**
  - o Create an Azure Virtual Machine (VM) with a system-assigned Managed Identity enabled.
  - o Assign the **Storage Blob Data Reader** role to the Managed Identity for "secure storage".

o Connect to the VM and verify that the Managed Identity can access blob data using Azure CLI or a pre-installed script.

## 1. Created a storage account named "securestorage2321" and added **Storage Blob Data Contributor** role

Name	Type	Role	Scope	Condition
dbrov.vladimir@gmail.com#EXT#@dbrovadminmrgmail... User	User	Owner	Subscription (Inherited)	None
dbrov.vladimir@gmail.com#EXT#@dbrovadminmrgmail... User	User	Owner	Subscription (Inherited)	None
Unable to find identity	Unknown	Owner	Subscription (Inherited)	None
Identity not found	Unknown	Owner	Subscription (Inherited)	None
Unable to find identity	Unknown	Owner	Subscription (Inherited)	None
Developers	Group	App Service Environment Contributor	Subscription (Inherited)	None
Key Vault Administrator	Role	Key Vault Administrator	Subscription (Inherited)	None
Key Vault Reader	Role	Key Vault Reader	Subscription (Inherited)	None
Developers	Group	Monitoring Contributor	Subscription (Inherited)	None
Reader	Role	Monitoring Contributor	Subscription (Inherited)	None
Admins	Group	Reader	Subscription (Inherited)	None
Storage Account Contributor	Role	Storage Account Contributor	Subscription (Inherited)	None
Storage Blob Data Contributor	Role	Storage Blob Data Contributor	This resource	Add

Verified upload and download blobs to the account.

Name	Modified	Access tier	Archive status	Blob type	Size	Lease state
blobTest.txt	1/21/2023, 4:43:08 PM	Hot (Inferred)		Block blob	9 B	Available

Another account:

The screenshot shows the Microsoft Azure Storage Accounts interface for a container named 'test-container'. The top navigation bar includes links for Home, Storage accounts, and securestorage2321 | Containers. The main content area displays the container's properties: Name (test-container), Type (Container), and a status message indicating no access key is present. A 'Search' bar is at the top left. Below it, a sidebar lists Overview, Diagnose and solve problems, Access Control (IAM), and Settings. The 'Overview' tab is selected. The main content area features a large red warning box about missing access keys. It includes a 'Search blobs by prefix (case-sensitive)' input field and a 'Show deleted blobs' toggle switch. At the bottom, there's a table header for blob management with columns: Name, Modified, Access tier, Archive status, Blob type, Size, and Lease state. The table body shows 'No results'.

Created VM and assigned the **Storage Blob Data Reader** role to the Managed Identity for "securestorage2321".

Access Control (IAM)					
	Name	Type	Role	Scope	Condition
Overview					
Activity log					
Tags					
Diagnose and solve problems					
Access Control (IAM)					
Data migration					
Events					
Storage browser					
Storage Mover					
Partner solutions					
Data storage					
Containers					
File shares					
Queues					
Tables					
Security + networking					
Data management					
Settings					
Monitoring					
Monitoring (classic)					
Automation					
Help					
Voxymir Dzmitry (dvozvuyadm@gmail.com#EXT#@dbrovadim@gmail.com) (Owner)					
Voxymir Dzmitry (dvozvuyadm@gmail.com#EXT#@dbrovadim@gmail.com) (User)					
Voxymir Dzmitry (dvozvuyadm@gmail.com#EXT#@dbrovadim@gmail.com) (User)					
Identity not found. Unable to find identity.					
Identity not found. Unable to find identity.					
Identity not found. Unable to find identity.					
Dereferences					
Group					
App Service Environment Contributor (I)					
Key Vault Administrator (I)					
Key Vault Reader (I)					
Developers					
Group					
Monitoring Contributor (I)					
Reader (I)					
Admins					
Group					
Storage Account Contributor (I)					
Storage Blob Data Contributor (I)					
Storage Blob Data Reader (I)					
MyWindowsVM					
Virtual Machine					

```
az login --identity
{
  "environment": "AzureCloud",
  "id": "0507be71-1145-4a77-bd8b-dfa9921ae5d",
  "isDefault": true,
  "managedIdentity": [],
  "name": "Azure subscription 1",
  "state": "Enabled",
  "tenantId": "0507be71-1145-4a77-be37-4b46-bf3a-14371b215538",
  "user": {
    "id": "0507be71-1145-4a77-be37-4b46-bf3a-14371b215538",
    "name": "live.com\dibrovladimir@gmail.com",
    "type": "user"
  }
}
```

```
Microsoft Azure          | Search resources, services, and docs (G+)
[?] Help                    | Copy
[?] Select to PowerShell  | Run in Bash | Manage File  | New session  | Editor  | Web preview  | Settings  | Help  |
[?] [x] storage blob download --account-name securestorage2221 --container-name test-container --name blobtest.txt --file blobtest.txt --auth-mode login
finished: [=====] 100.0%
{
  "container": "test-container",
  "content": "",
  "contentMD5": null,
  "deleted": false,
  "encryptionKey": null,
  "encryptionKeySha256": null,
  "encryptionScope": null,
  "lastAccessed": null,
  "hasVersioningOnly": null,
  "immutabilityPolicy": {
    "expiryTime": null,
    "permblob": null
  },
  "isAppendableblob": null,
  "isServerVersion": null,
  "lastAccessedOn": null,
  "metadata": {},
  "name": "blobtest.txt",
  "objectReplicationDestinationPolicy": null,
  "objectReplicationSourceProperties": [],
  "populated": false,
  "spacedBlockCommitmentBlockCount": null,
  "blobList": null,
  "blobName": null,
  "blobInferred": null,
  "blobType": "Blockblob",
  "blobLength": null,
  "contentLanguage": "Mytes None-None/0",
  "contentSettings": {}
}
```

```

    "tags": null,
    "versionId": null
}
volodymyr [ ~ $] cat blobTest.txt
Xxexexexevolodymyaz storage blob upload --account-name securestorage2321 --container-name test-container --name blobTest.txt --file blobTest.txt --auth-mode login

```

```

volodymyr [ ~ $] az storage blob upload --account-name securestorage2321 --container-name test-container --name blobTest.txt --file blobTest.txt --auth-mode login --overwrite
Finished[########################################] 100.0000%
{
  "client_request_id": "651084f4-d818-11ef-a104-00155ddd6d38",
  "content_md5": "6ly2thHlBVbIXKasgwP8Gg==",
  "date": "2025-01-21T16:54:35+00:00",
  "encryption_key_sha256": null,
  "encryption_scope": null,
  "etag": "'0x8000A3C494B3E45'",
  "lastModified": "2025-01-21T16:54:35+00:00",
  "request_id": "4801dd52-e0fe-0005-1a25-6c08c4000000",
  "request_server_encrypted": true,
  "version": "2022-11-02",
  "version_id": null
}
volodymyr [ ~ $] az account show
{
  "environmentName": "AzureCloud",
  "homeTenantId": "cd0728c4-be37-4b46-bf3a-14371b215538",
  "id": "507cbe71-1145-4a77-1bd8-d6fa9921aed5",
  "isDefault": true,
  "managedByTenants": [],
  "name": "Azure subscription 1",
  "state": "Enabled",
  "tenantId": "cd0728c4-be37-4b46-bf3a-14371b215538",
  "user": {
    "cloudshellId": true,
    "name": "live.com#dibrov.vladimir@gmail.com",
    "type": "user"
  }
}
volodymyr [ ~ $]

```

## Practical Task 8: Creating and Querying an Azure SQL Database

Learn to create and query an Azure SQL Database using the Azure portal.

### Requirements:

1. Create an Azure SQL Database named "test-db" in a new logical SQL server.
2. Set the pricing tier to the free tier for cost optimization.
3. Use the Query Editor in the Azure portal to create a table named "Products" with columns for ID, Name, and Price.
4. Insert a few sample records into the "Products" table using a SQL query.
5. Query the table to display all records and verify the data.

The screenshot shows the Azure portal interface for creating and managing an Azure SQL Database. On the left, the 'SQL databases' blade is open, showing a list of databases under 'Default Directory'. A new database 'test-db' is being created under 'test-sql-server-12345/test-db'. The 'Query editor (preview)' tab is active, displaying a SQL script to create a 'Products' table and insert three sample records. The results pane shows the three inserted rows: Laptop (ID 1, Price 999.99), Phone (ID 2, Price 499.99), and Tablet (ID 3, Price 299.99).

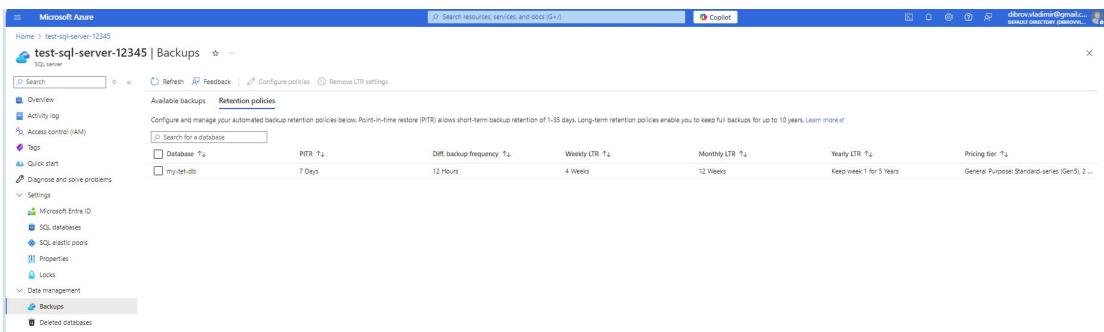
ID	Name	Price
1	Laptop	999.99
2	Phone	499.99
3	Tablet	299.99

## Practical Task 9: Deploying an Azure SQL Database with Automated Backup Configuration

Set up an Azure SQL Database and configure automated backups. **Requirements:**

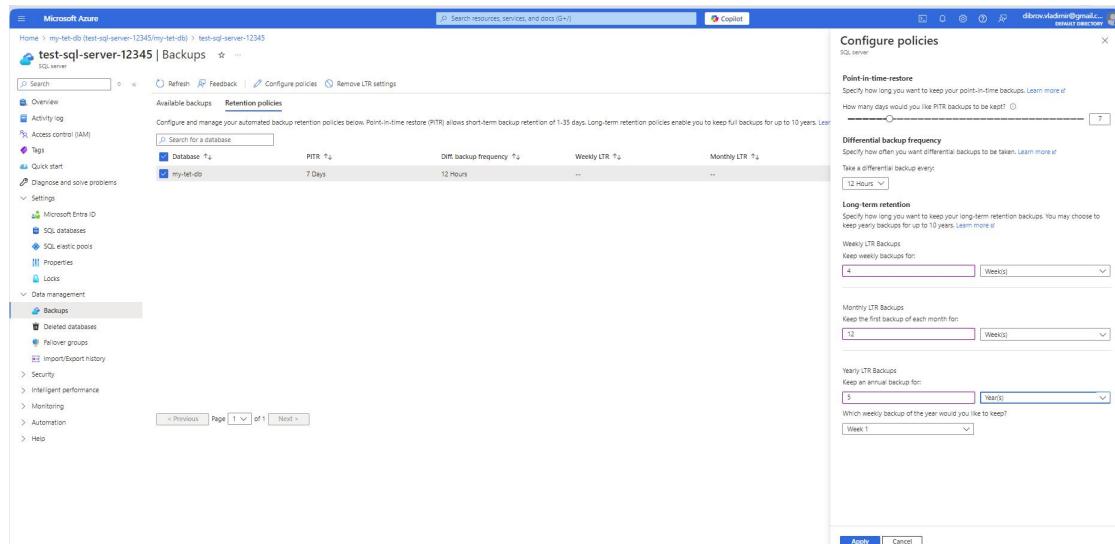
1. Create an Azure SQL Database in a new or existing resource group.
2. Choose the desired service tier (e.g., Basic or General Purpose).
3. Enable and configure long-term backup retention for the database.
4. Use the Azure portal to verify backup settings and review available restore points.
5. Test the restore process by creating a new database from a backup

## 1.Created an Azure SQL Database and selected the desired service tier - General Purpose

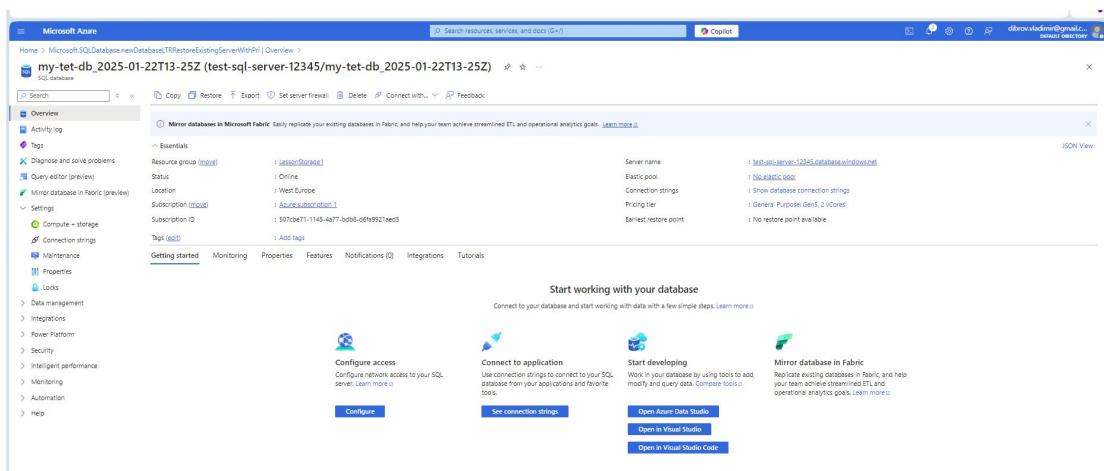


The screenshot shows the Azure portal interface for managing a database. The left sidebar includes options like Overview, Activity log, Access control (IAM), Tags, Quick start, Diagnose and solve problems, Settings, Microsoft Entra ID, SQL databases, SQL elastic pools, Properties, Locks, Data management, and Backups. The 'Backups' section is currently selected. Under 'Retention policies', there are two rows: one for 'Available backups' (PITR) and one for 'my-tet-db' (LTR). The LTR row shows a PITR of 7 days, a Diff. backup frequency of 12 hours, and a Weekly LTR of 4 weeks. The Monthly LTR is set to 12 weeks, and the Yearly LTR is set to keep week 1 for 5 years. The Pricing tier is listed as General Purpose Standard-series (Gen5), 2 vCores.

## 2.Enabled and configure policies for restoring long-term backup retention for the database



This screenshot shows the 'Configure policies' dialog for the database. It has tabs for 'Point-in-time restore' and 'Long-term retention'. In the 'Point-in-time restore' section, it specifies how many days PITR backups should be kept (7 days). In the 'Long-term retention' section, it specifies how often differential backups are taken (12 hours) and how long LTR backups are kept (weekly for 4 weeks, monthly for 12 weeks, and yearly for 5 years). Buttons for 'Apply' and 'Cancel' are at the bottom.



The screenshot shows the 'Overview' page for the database. The left sidebar includes Overview, Activity log, Tags, Diagnose and solve problems, Query editor (preview), Mirror database in Fabric (preview), Settings, Compute + storage, Connection strings, Maintenance, Properties, Data management, Integrations, Power Platform, Security, Intelligent performance, Monitoring, Automation, and Help. The 'Getting started' tab is selected. Below the main content, there are four cards: 'Configure access' (Configure network access to your SQL server), 'Connect to application' (Use connection string to connect to your SQL database from your applications and favorite tools), 'Start developing' (Work in your database by using tools to add, modify, and query data. Compare tools), and 'Mirror database in Fabric' (Replicate existing databases in Fabric, and help your team achieve streamlined ETL and open-source analytics goals).

### 3. Tested using rule in Monitoring

### Practical Task 10: Getting Started with Cosmos DB

Set up and explore Cosmos DB by creating a database, managing data, querying, and testing key features like consistency and global distribution.

#### Requirements:

##### 1. Create a Cosmos DB Account:

- o Create a new Cosmos DB account in the Azure portal using the Core (SQL) API.
- o Select a region for the account and use the default settings.
- o Review key features like throughput, consistency levels, and global distribution.
- o Note down the primary and secondary keys for the account.

##### 2. Set Up a Database and Container:

- o Create a database named "SampleDB" in the Cosmos DB account.
- o Add a container named "Items" with the partition key set to /category.
- o Use the default throughput settings for the database and container.

##### 3. Insert Data Using Data Explorer:

- o Open the Data Explorer in the Azure portal.
- o Insert at least three sample JSON documents into the "Items" container. Example:

```
{
  "id": "1",
  "name": "Smartphone",
  "category": "Electronics",
  "price": 699
}
```

- o Verify that the documents are successfully added and visible in the Data Explorer.

#### 4. Query Data in Cosmos DB:

- o Use SQL-like queries in the Data Explorer to retrieve data:

- Retrieve all documents: `SELECT * FROM c`
- Retrieve specific documents: `SELECT * FROM c WHERE c.category = 'Electronics'`

- o Observe how the partition key affects query results.

#### 5. Configure and Test Consistency Levels:

- o Review the available consistency levels in the Cosmos DB account: Eventual, Session, Consistent Prefix, Bounded Staleness, Strong.

- o Set the account-level consistency to "Session."

- o Perform a query or data operation and observe the impact on performance and latency.

#### 6. Enable Global Distribution and Test Replication:

- o Enable multi-region replication by adding at least one additional region to the Cosmos DB account.

- o Insert new data into the primary region and verify that it replicates to the secondary region.

- o Perform a manual failover and verify that the secondary region becomes the primary.

- o Test data consistency and latency after failover.

### 1. Created a Cosmos DB Account ->Configured the database and container ->Insert Data Using Data Explorer

**Overview** page details:

- Status: Online
- Resource group (Region): LessonStorage
- Subscription (Region): Azure Subscription 1
- Subscription ID: 507ce71-1145-477-b0b8-d6fe9927ae5
- Total throughput limit: 1000 RU/s
- Read Locations: West Europe
- Write Locations: West Europe
- URI: https://mycosmosdbaccount12.documents.azure.com:443/
- Free Tier Discount: Opted In
- Capacity mode: Provisioned throughout

**Monitoring** section shows request distribution over 24 hours:

Request Type	Count
Total	1000
HTTP 2xx	800
HTTP 400	100
HTTP 401	100
HTTP 403	100

**Estimated Cost (Hourly)** chart shows throughput and storage usage.

**Data Explorer** page details:

- Container: Items
- Items:

  - 1. Electronics, name: Smartphone, price: 699
  - 2. Electronics, name: Laptop, price: 1200
  - 3. Furniture, name: Table, price: 150

## 2.Query Data in Cosmos DB

```

1 SELECT * FROM c
2
[{"id": "1", "name": "Smartphone", "category": "Electronics", "price": 699, "_rid": "tliYAM169vIBAAAAAAA==", "_self": " dbs/tliYAA==/colls/tliYAM169vI/docs/tliYAM169vIBAAAAAAA==/", "_etag": "\\"66dded385-0000-0000-0000-67911ee70000\\\"", "_attachments": "attachments/", "_ts": 1737563879}, {"id": "2", "name": "Laptop", "category": "Electronics", "price": 1200, "_rid": "tliYAM169vICAAAAAAA==", "_self": " dbs/tliYAA==/colls/tliYAM169vI/docs/tliYAM169vICAAAAAAA==/", "_etag": "\\"66065b93-0000-0000-0000-67911fce0000\\\"", "_attachments": "attachments/", "_ts": 1737564118}, {"id": "3", "name": "Table", "category": "Furniture", "price": 150, "_rid": "tliYAM169vIDAAAAAAA==", "_self": " dbs/tliYAA==/colls/tliYAM169vI/docs/tliYAM169vIDAAAAAAA==/", "_etag": "\\"66066294-0000-0000-0000-67911fe10000\\\"", "_attachments": "attachments/", "_ts": 1737564129}]

```

```

1 SELECT * FROM c WHERE c.category = 'Electronics'
2
[{"id": "1", "name": "Smartphone", "category": "Electronics", "price": 699, "_rid": "tliYAM169vIBAAAAAAA==", "_self": " dbs/tliYAA==/colls/tliYAM169vI/docs/tliYAM169vIBAAAAAAA==/", "_etag": "\\"66dded385-0000-0000-0000-67911ee70000\\\"", "_attachments": "attachments/", "_ts": 1737563879}, {"id": "2", "name": "Laptop", "category": "Electronics", "price": 1200, "_rid": "tliYAM169vICAAAAAAA==", "_self": " dbs/tliYAA==/colls/tliYAM169vI/docs/tliYAM169vICAAAAAAA==/", "_etag": "\\"66065b93-0000-0000-0000-67911fce0000\\\"", "_attachments": "attachments/", "_ts": 1737564118}, {"id": "3", "name": "Table", "category": "Furniture", "price": 150, "_rid": "tliYAM169vIDAAAAAAA==", "_self": " dbs/tliYAA==/colls/tliYAM169vI/docs/tliYAM169vIDAAAAAAA==/", "_etag": "\\"66066294-0000-0000-0000-67911fe10000\\\"", "_attachments": "attachments/", "_ts": 1737564129}]

```

### 1. Configured and Test

**Eventual – fastest option with possible delays.**

**Session – default to balance performance and consistency.**

**Consistent Prefix – order of changes is partially guaranteed.**

**Bounded Staleness – controlled update latency.**

**Strong – full consistency, but with higher performance costs.**

The screenshot shows the Microsoft Azure Cosmos DB Overview page for the account 'mycosmosdbaccount12'. The left sidebar contains navigation links like Overview, Activity log, Access control (IAM), Tags, Diagnose and solve problems, Cost Management, Quick start, Data Explorer, Settings, and more. The main content area has tabs for STRONG, BOUNDED STALENESS, SESSION (which is selected), CONSISTENT PREFIX, and EVENTUAL. A note states: 'Session consistency is most widely used consistency level both for single region as well as, globally distributed applications.' Below this is a world map showing data centers. To the right, there are four musical staff diagrams illustrating reads and writes across regions:

- West Europe Writes in Session A
- West Europe Reads in Session A
- West Central US Reads in Session A
- North Central US Reads in Session B

The screenshot shows the Microsoft Azure Cosmos DB Data Explorer page for the same account. The left sidebar includes Data Explorer under Settings. The main area features a search bar, a New SQL Query button, and tabs for Home, Items, and Items.Query 1. The Items.Query 1 tab displays a query: 'SELECT \* FROM E'. Below the query, the 'Query Stats' section provides detailed metrics:

Metric	Value
Request Charge	2.29 RU/s
Showing Results	1 - 3
Retrieved document count	3
Retrieved document size	480 bytes
Output document count	3
Output document size	657 bytes
Index hit document count	3
Index lookup time	0 ms
Document load time	0.02 ms
Query engine execution time	0.01 ms
System function execution time	0 ms
User defined function execution time	0 ms
Document write time	0.01 ms
Round Trips	1

At the bottom, it says '↓ Per-partition query metrics (CSV)'

**Microsoft Azure**

Home > mycosmosdbaccount12

### mycosmosdbaccount12 | Data Explorer

Azure Cosmos DB account

Search   Save Discard

- [Overview](#)
- [Activity log](#)
- [Access control \(IAM\)](#)
- [Tags](#)
- [Diagnose and solve problems](#)
- [Cost Management](#)
- [Quick start](#)
- Data Explorer**
- [Settings](#)
  - [Features](#)
  - [Replicate data globally](#)
  - [Default consistency](#)
  - [Backup & Restore](#)
  - [Networking](#)
  - [CORS](#)
  - [Dedicated Gateway](#)
  - [Keys](#)
  - [Advisor Recommendations](#)

Home Items Scale & ... Items.Items Items.Query 1

Scale	Settings	Indexing Policy	Partition Keys (preview)	Computed Properties
1	2	3 "indexingMode": "consistent", 4 "automatic": true, 5 "includedPaths": [ 6     { 7         "path": "//*" 8     } 9 ], 10 "excludedPaths": [ 11     { 12         "path": "/\"_etag\"/?" 13 } 14 ]		

Execute Query Save Query Download Query View

Home Items Scale & ... Items.Items Items.Query 1

```
1 SELECT * FROM c WHERE c.category = "Electronics"
```

Results Query Stats

Metric	Value
Request Charge	2.84 RUs
Showing Results	1-2
Retrieved document count	2
Retrieved document size	324 bytes
Output document count	2
Output document size	591 bytes
Index hit document count	2
Index lookup time	0.0900000000000001 ms
Document load time	0.08 ms
Query engine execution time	0.0300000000000002 ms
System function execution time	0 ms
User defined function execution time	0 ms
Document write time	0.01 ms
↓ Per-partition query metrics (CSV)	

Microsoft Azure

Home > mycosmosdbaccount12

### mycosmosdbaccount12 | Data Explorer

Azure Cosmos DB account

Search  Enable Azure Synapse Link

New SQL Query Open Query New Stored Procedure

Overview Activity log Access control (IAM) Tags Diagnose and solve problems Cost Management Quick start Data Explorer Settings Features Replicate data globally Default consistency Backup & Restore Networking CORS Dedicated Gateway Keys Advisor Recommendations Microsoft Defender for Cloud Identity Locks

+ New Container

Home SampleDB Items

Items Scale & ... Indexing Policy Partition Keys (preview) Computed Properties

You have not saved the latest changes made to your indexing policy. Please click save to confirm the changes.

```
1 {
2   "includedPaths": [
3     {
4       "path": "/category/?",
5       "indexes": [
6         {
7           "kind": "Range",
8           "dataType": "String"
9         }
10      ]
11    }
12  ]
13 }
```

Microsoft Azure

Home > mycosmosdbaccount12

### mycosmosdbaccount12 | Data Explorer

Azure Cosmos DB account

Search  Execute Query Save Query Download Query View

Overview Activity log Tags Diagnose and solve problems Cost Management Quick start Data Explorer Settings Features Replicate data globally Default consistency Backup & Restore Networking CORS Dedicated Gateway Keys Advisor Recommendations Microsoft Defender for Cloud Identity Locks Integrations Containers Monitoring Automation Help

+ New Container

Home SampleDB Items

Items Scale & Settings

Results Query Stats

Metric	Value
Request Charge	2.84 RUs
Showing Results	1 - 2
Retrieved document count	2
Retrieved document size	324 bytes
Output document count	2
Output document size	591 bytes
Index hit document count	2
Index lookup time	0.11 ms
Document load time	0.02 ms
Query engine execution time	0.0300000000000002 ms
System function execution time	0 ms
User defined function execution time	0 ms
Document write time	0.01 ms

↓ Per-partition query metrics (CSV)